Microsoft

Reactor

STOCKHOLM

This session will commence shortly

We are constantly striving to create excellent content
and would appreciate if you could take this brief survey.

Survey Link: https://aka.ms/Reactor/Survey

Please enter the event code 12788 at the start of survey

Speaker Slide:

Massimo Bonanni

Azure Technical Trainer @ Microsoft

I spend my time to help customers to empower their Azure skills to achieve more and leverage the power of Azure in their solutions.
I'm also a technical speaker both for local and international events and a user-group guy.
I founded Aa couple of communities in Italy and collaborated with most of the Italian communities.
Finally, I is also passionate about biking, reading, and dogs!!

Empower every Azure Function to achieve more!!

Massimo Bonanni

Azure Technical Trainer @ Microsoft

meetup.com/Microsoft-Reactor-Stockholm/

Our Code of Conduct

Microsoft is dedicated to empowering every person and every organization on the planet to achieve more.

This includes Microsoft Reactor events where we seek to provide a respectful, friendly, professional experience for everyone, regardless of gender, sexual orientation, physical appearance, disability, age, race or religion.

We do not tolerate any behaviour that is harassing or degrading to any individual, in any form. Individuals are responsible for knowing and abiding by these standards. We encourage everyone to assist in creating a welcoming and safe environment.

Be aware of others

Be friendly and patient

Be welcoming and respectful

Be open to all questions and viewpoints

Be understanding of differences

Be kind and considerate to others

What is serverless?

Full abstraction of servers

Developers can just focus on their code—there are no distractions around server management, capacity planning, or availability.

Instant, event-driven scalability

Application components react to events and triggers in near real-time with virtually unlimited scalability; compute resources are used as needed.

Pay-per-use

Only pay for what you use: billing is typically calculated on the number of function calls,
code execution time, and memory used.*

What are Azure Functions?

Events

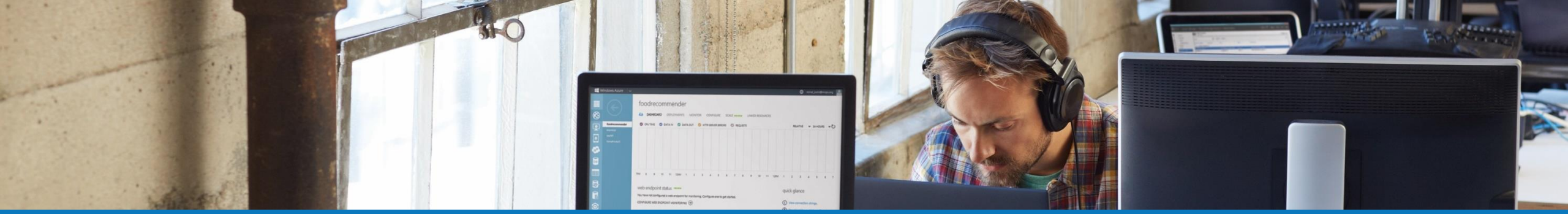
React to timers, HTTP, or events from your favorite Azure services, with more on the way

Code

Author functions in C#, F#, Node.JS, Java, and more

Outputs

Send results to an ever-growing collection of services



Boost development efficiency

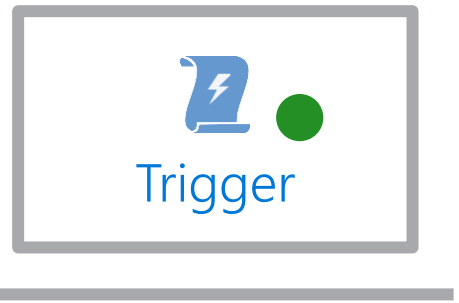
Trigger object

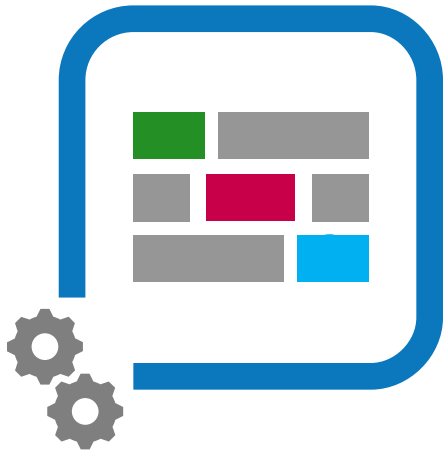
Your code

Input object

Output object

Input Binding





Output Binding

Anatomy of an Azure Function

```
[FunctionName("CopyQueueMessage")]
public static void Run(
    [QueueTrigger("myqueue-items-source")] string myQueueItem,
    [Queue("myqueue-items-destination")] out string myQueueItemCopy,
    ILogger log)
{
    log.LogInformation($"CopyQueueMessage function processed: {myQueueItem}");
    myQueueItemCopy = myQueueItem;
}
```

Trigger Attribute

Binding Attribute

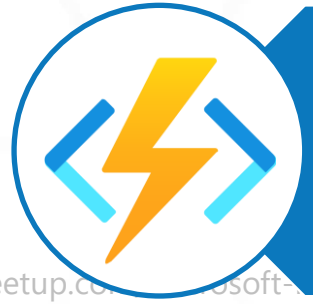
Trigger Payload

Binding Payload

Extend triggers and bindings

All Triggers and Bindings (except for HTTPTrigger and Timer Trigger) are available as **external packages**.

The **Azure Functions SDK** is based on the **Azure WebJobs SDK** and inherits the extension SDK from it.



An extension is a class that implements the **`IExtensionConfigProvider`** interface.

Azure Functions lifecycle phases

Startup

The runtime executes this phase only when the host starts.

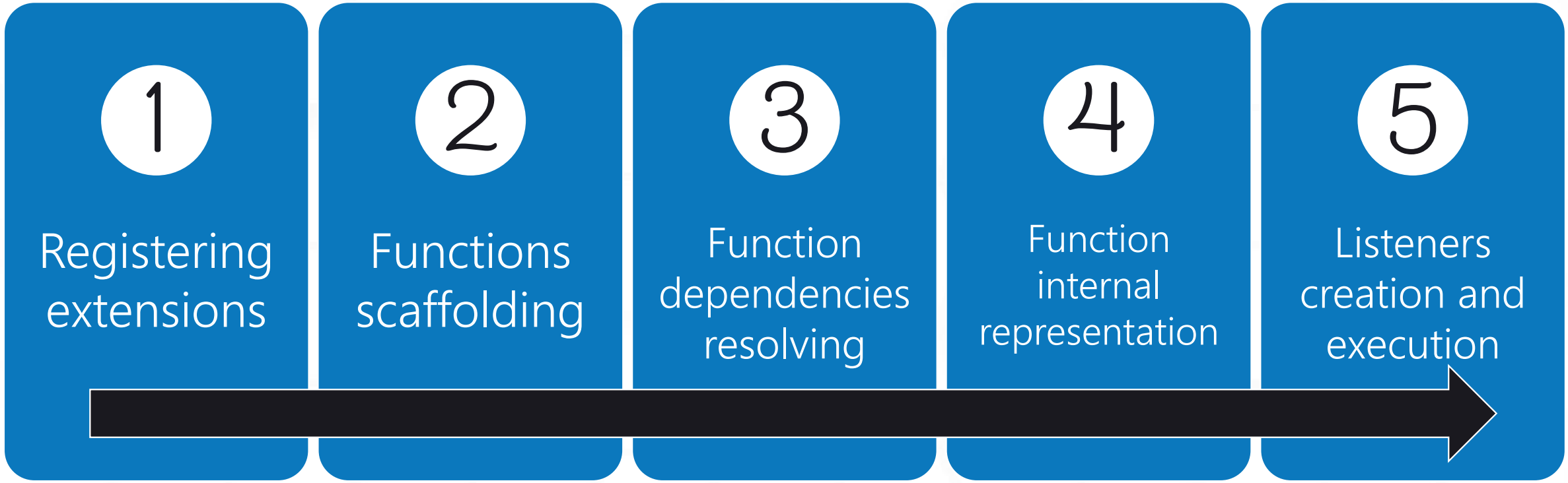
The runtime registers the built-in binding (TimerTrigger and HttpTrigger).

You must register your custom extensions.

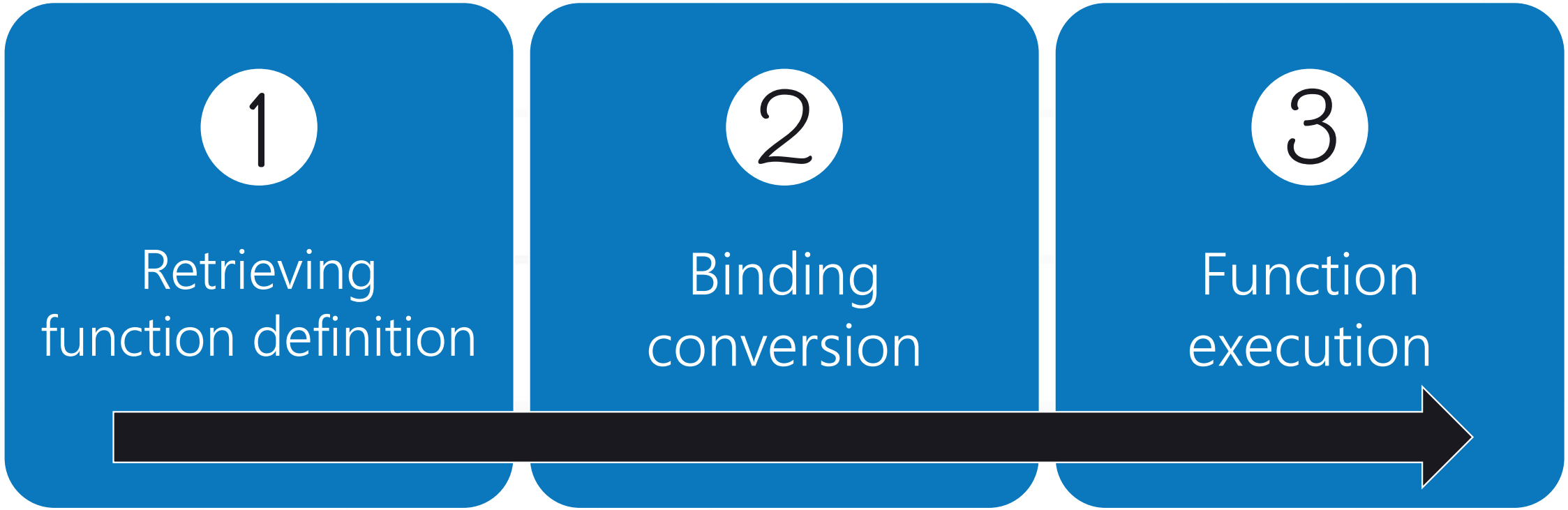
Runtime

The runtime executes this phase every time a function is triggered by an event.

The Startup phase



The Runtime phase



Trigger in deep

Classes involved in a Trigger

TriggerAttribute

TriggerConfigProvider

TriggerBindingProvider

TriggerBinding

TriggerListener

Classes involved in a Trigger

TriggerAttribute

Decorates an argument of a method to identify the trigger

TriggerConfigProvider

TriggerBindingProvider

TriggerBinding

TriggerListener

Classes involved in a Trigger

Decorates an argument of a method to identify the trigger

TriggerAttribute

Define the extension (implementing the IExtensionConfigProvider)

TriggerConfigProvider

TriggerBindingProvider

TriggerBinding

TriggerListener

Classes involved in a Trigger

Decorates an argument of a method to identify the trigger

TriggerAttribute

Define the extension (implementing the IExtensionConfigProvider)

TriggerConfigProvider

Factory class for creating the actual binding object

TriggerBindingProvider

TriggerBinding

TriggerListener

Classes involved in a Trigger

Decorates an argument of a method to identify the trigger

TriggerAttribute

Define the extension (implementing the IExtensionConfigProvider)

TriggerConfigProvider

Factory class for creating the actual binding object

TriggerBindingProvider

Binding object, creates the actual listener

TriggerBinding

TriggerListener

Classes involved in a Trigger

Decorates an argument of a method to identify the trigger

TriggerAttribute

Define the extension (implementing the IExtensionConfigProvider)

TriggerConfigProvider

Factory class for creating the actual binding object

TriggerBindingProvider

Binding object, creates the actual listener

TriggerBinding

It reacts to events and executing the function

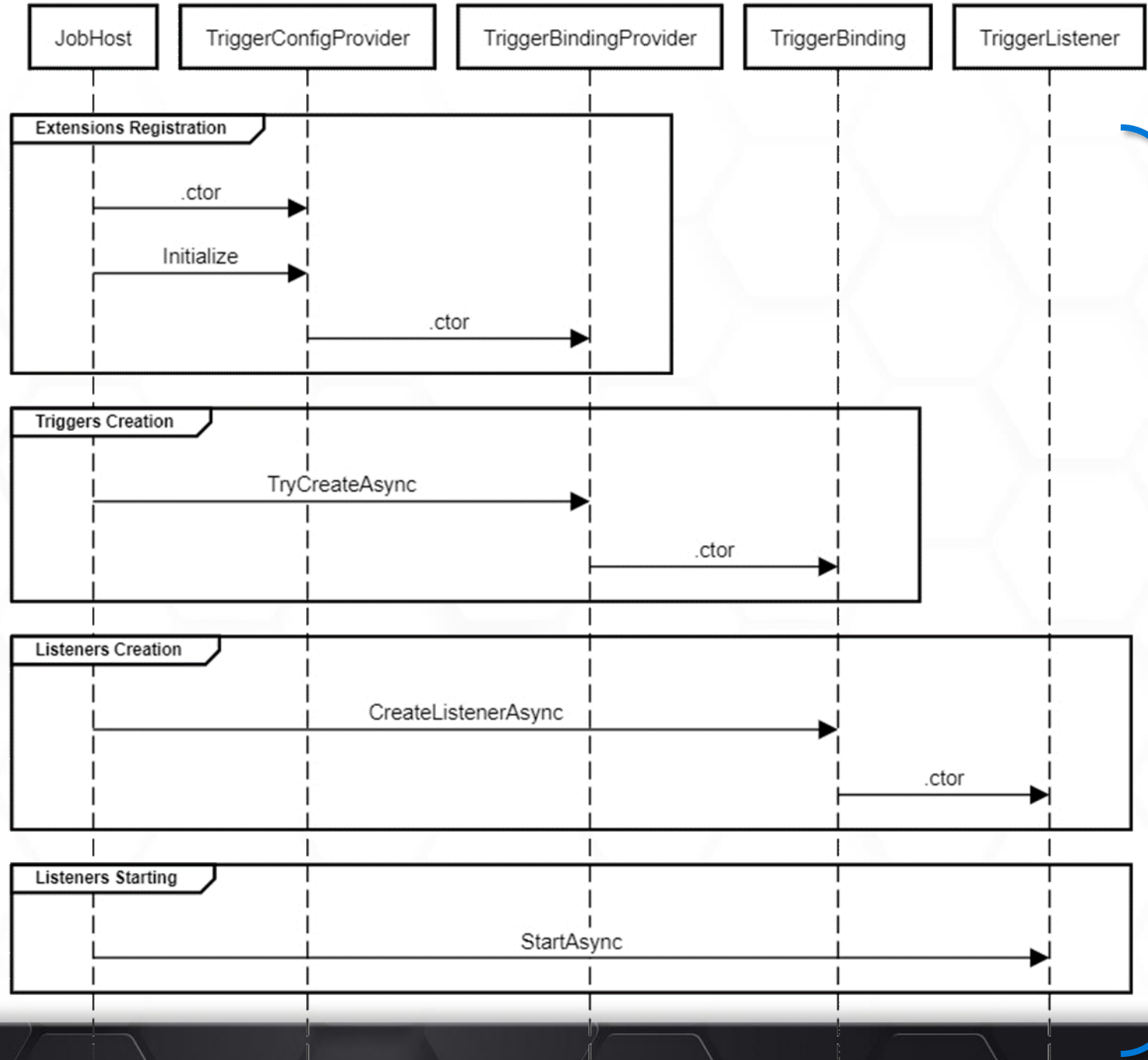
TriggerListener

The runtime creates the factory for binding

The runtime uses the factory to create the binding instance

The runtime uses binding class to create the listener

The runtime starts the listener



Startup Phase

Weather Custom Trigger

Binding in deep

Classes involved in a Binding

BindingAttribute

BindingConfigProvider

BindingConverter

Binding class

Classes involved in a Binding

BindingAttribute

Decorates an argument of a method to identify the binding

BindingConfigProvider

BindingConverter

Binding class

Classes involved in a Binding

Decorates an argument of a method to identify the binding

BindingAttribute

Define the extension (implementing the IExtensionConfigProvider)

BindingConfigProvider

BindingConverter

Binding class

Classes involved in a Binding

BindingAttribute

Decorates an argument of a method to identify the binding

BindingConfigProvider

Define the extension (implementing the IExtensionConfigProvider)

BindingConverter

Creates the actual binding class for the binding

Binding class

Classes involved in a Binding

BindingAttribute

Decorates an argument of a method to identify the binding

BindingConfigProvider

Define the extension (implementing the IExtensionConfigProvider)

BindingConverter

Creates the actual binding class for the binding

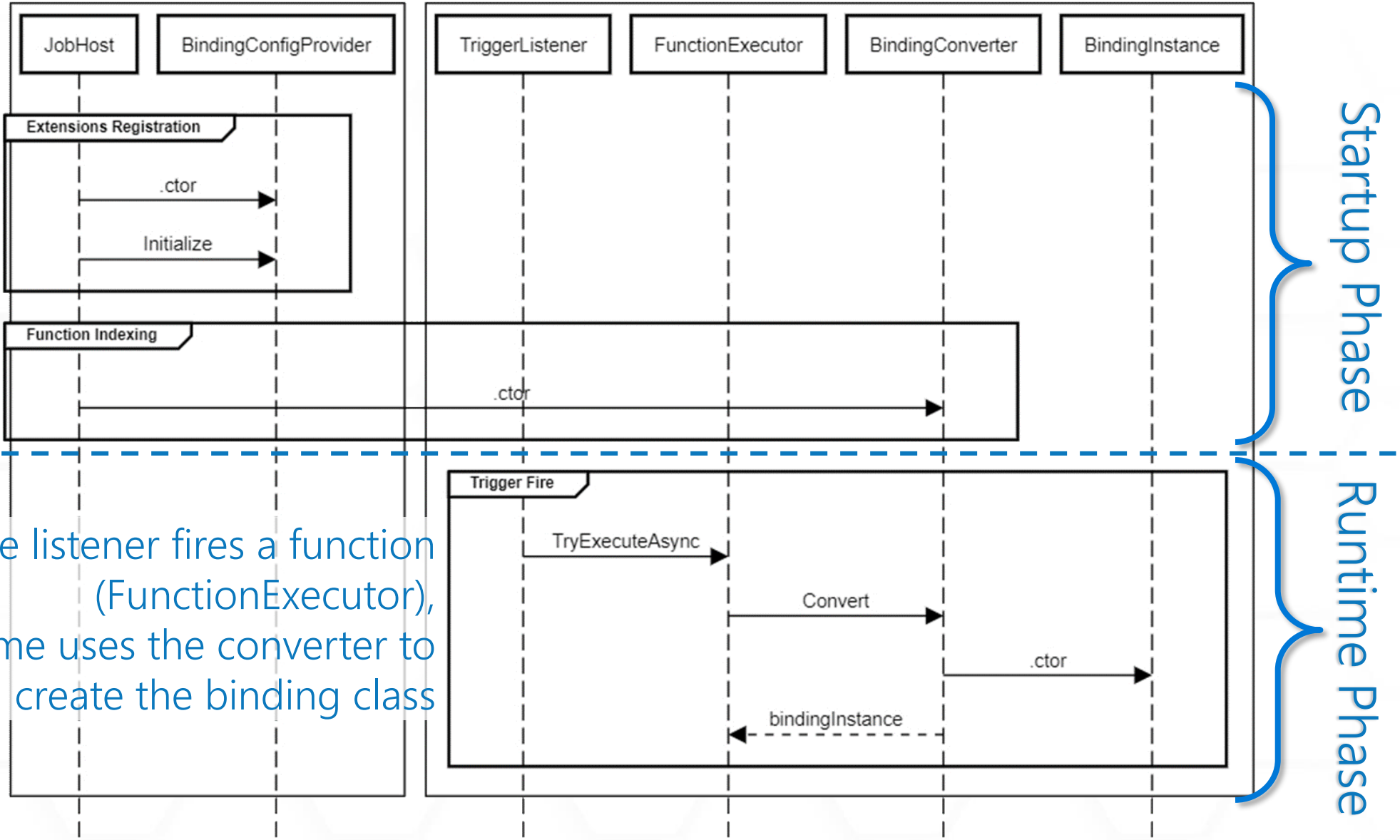
Binding class

The class that actually binds to the data source

The runtime creates the extension

The runtime creates the converter

When the listener fires a function (FunctionExecutor), the runtime uses the converter to create the binding class



Twitter Custom Binding

Takeaways

Implementing your own triggers and bindings allows you to abstract the data source with respect to the Azure Function code.

You pay for duration and memory occupation of your function.

Your code must be efficiently and avoid to load assembly that you don't use.

Trigger listener is one of the most important classes for scalability and performance: write it once and the best you can!!!

Thanks for your attention!!!!

Massimo Bonanni

Azure Technical Trainer

massimo.bonanni@microsoft.com
@massimobonanni

Connect with me on LinkedIn

linkedin.com/in/massimobonanni/

meetup.com/Microsoft-Reactor-Stockholm/

http://bit.ly/MasteringServerless

Join our community

meetup.com/Microsoft-Reactor-Stockholm/

@MSFTReactor

http://www.youtube.com/c/MicrosoftReactor

ReactorStockholm@microsoft.com

Microsoft Reactor at Epicenter,
Master Samuelsgatan 36, 5th floor,
111 57 Stockholm Sweden

Questions? ReactorStockholm@microsoft.com

References

- Azure Functions Documentation
  https://docs.microsoft.com/en-US/azure/azure-functions/
- Azure Functions Code Samples
  https://azure.microsoft.com/en-us/resources/samples/?service=functions&sort=0
- Azure Updates
  https://azure.microsoft.com/en-us/roadmap/?category=compute
- Demo AccuWeather Trigger / Twitter Binding – GitHub
  https://github.com/massimobonanni/AzureFunctionsSamples
- Demo SQL Trigger/Binding – GitHub
  https://github.com/massimobonanni/SQLServerless

Photo by Jaredd Craig on Unsplash